# Why is getting the HP_HASHSIZE so weird?

devblogs.microsoft.com/oldnewthing/20160128-00

January 28, 2016

Raymond Chen

A comment on the documentation page for `CryptGetHashParam` notes that the “obvious” way to get the `HP_HASHSIZE` is incorrect.

```
// Version 1: wrong.
DWORD size = 0;
if (CryptGetHashParam(hash, HP_HASHSIZE, nullptr, &size, 0)) ...

// Version 2: right.
DWORD size;
DWORD bufferSize = sizeof(size);
if (CryptGetHashParam(hash, HP_HASHSIZE, &size, &bufferSize, 0)) ...
```

What’s going on here? I mean, the documentation says that if you want to get the size of a parameter, you pass `nullptr` for the buffer, and the `DWORD*` parameter gets the size of the buffer. So if I want to get the hash size, I should pass `nullptr` for the buffer, and the `DWORD*` parameter gets the size of the hash. But it doesn’t. It always returns 4. What’s going on?

What’s going on is that you are working at the wrong level of indirection. The code in version 1 is not asking for the size of the hash. It’s asking for the size of the `HP_HASHSIZE`. In other words, you’re asking for the size of the *size*. Since `HP_HASHSIZE` is a `DWORD`, its size is 4. You then need to follow up with the code in version 2, which allocates a buffer of size 4 and asks for it to be filled in with the `HP_HASHSIZE`.

A third way to get the size of the hash is to ignore `HP_HASHSIZE` completely and go straight for the `HP_HASHVAL`:

```
// Version 3: righter
DWORD hashSize = 0;
if (CryptGetHashParam(hash, HP_HASHVAL, nullptr, &hashSize, 0)) ...
```

I don’t know why the crypto folks bothered to have a `HP_HASHSIZE` parameter. Adding it only created confusion.

Raymond Chen

## Follow